UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/994,483	11/26/2001	Michael A. Eidson	10559-515001/ P12419	7245

20985 7590 03/22/2006

FISH & RICHARDSON, PC
P.O. BOX 1022
MINNEAPOLIS, MN 55440-1022

EXAMINER	
RIZZUTO, KEVIN P	
ART UNIT	**PAPER NUMBER**
2183	

DATE MAILED: 03/22/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

PTO-90C (Rev. 10/03)

Office Action Summary	Application No.	Applicant(s)	
	09/994,483	EIDSON ET AL.	
	Examiner	Art Unit	
	Kevin P Rizzuto	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on 09 January 2006.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-42 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-42 is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on 12 February 2002 is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-42 have been examined.

2. Acknowledgement of papers filed: pre-appeal brief on 1/9/2006. The papers filed have been placed on record.

3. Applicant's arguments, see Pre-Appeal Brief, filed 1/9/06, with respect to the rejection(s) of claim(s) 1-42 under 35 U.S.C. 103 have been fully considered and are persuasive. Therefore, the rejections have been withdrawn. However, upon further consideration, a new ground(s) of rejection is made below and prosecution is hereby reopened.

Specification

4. The objection to the title of the invention is withdrawn.

New Claim Rejections - 35 USC § 102

5. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

6. Claims 1, 7, 15, 17, 19, 21, 23 and 25 are rejected under 35 U.S.C. 102(b) as being anticipated by Takase, U.S. Patent 6,289,473.

7. As per claim 1, Takase teaches a method comprising:

-In response to a first signal indicating the execution of a breakpoint by a processor (CPU 11) suspending execution of a peripheral ("each module") and saving the state of the peripheral: [Col. 8, line 48 to col. 9, line 19 describe the breakpoint and the signal that indicates a breakpoint has occurred. The state of the module is first saved (col. 9, lines 23-26), and then it outputs a stop notice signal, which indicates the module is stopped and the state has been kept, and then the debugger continues with the execution of the breakpoint. (Column 9, lines 20-42.)

-Continuing execution of the breakpoint by the processor in response to a second signal indicating that the state of the peripheral has been saved: [The state of the module is first saved (col. 9, lines 23-26), and then it outputs a stop notice signal, which indicates the module is stopped and the state has been kept, and then the debugger continues with the execution of the breakpoint. (Column 9, lines 20-42.) Further explanation is found on col. 11, line 1 to col. 12, line 20.]

-And restoring the saved state of the peripheral in response to a third signal indicating that execution of the breakpoint by the processor has been completed: [Col. 12, lines 21 to col. 13, line 27 and specifically, col. 13, lines 1-8.]

8. As per claims 7, 15, 17, 19, 21, 23 and 25, given the similarities between claim 1 and claims 7, 15, 17, 19, 21, 23 and 25, the arguments as stated for the rejection of claim 1 also apply to claims 7, 15, 17, 19, 21, 23 and 25.

New Claim Rejections - 35 USC § 103

9. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

10. Claims 1-27, 29-30, 32-33, 35-36 and 38-40 and 42 are rejected under 35 U.S.C. 103(a) as being unpatentable over Fowler, U.S. Patent 4,502,116 in view of Takase, U.S. Patent 6,289,473.

11. As per claim 1, Fowler teaches a method comprising:

-In response to a first signal (hardware or software induced interrupt) indicating the execution of a breakpoint by a processor (test utility system, not explicitly shown in drawings), suspending execution of a peripheral and saving the state of the peripheral (target subsystem): (Column 1, lines 26-31, Column 2, lines 59 to column 3, line 12, Steps 1-4 in Column 9, line 42-65)

-Continuing execution of the breakpoint by the processor after the state of the peripheral has been saved: (Step 5 in Column 9, line 66 to Column 10, line 22 indicates the peripheral's state has been saved, and afterwards in step 6, the utility, in response to the saved state being completed, continues execution by carrying out the appropriate action, which as stated in column 2, lines 6-25 step 3 includes means to display and modify registers and column 3, lines 7-12 and column 1, lines 26-31. Since step 6 is the next sequential step in the synchronized process in Column 9, line 42-65 and includes

modifying registers, it is inherent the saving of the state of the peripheral must be complete in step 5 before step 6 is started)

-And restoring the saved state of the peripheral in response to a third signal indicating that execution of the breakpoint by the processor has been completed: (The Test Utility Processor initiates a return to the application program in step 7 (shown in column 9, lines 42-65) via "an assert resume" portion of its software which also indicates the breakpoint has been completed. In the next step (step 8), after the initiation in step 7, the state of the subsystem is restored. (Column 2, lines 66-68 and Column 9, lines 42-65).

12. While Fowler teaches that the context/state of the target processor is saved after a breakpoint and before each individual processor is debugged/monitored, Fowler does not teach that the processor which initiates the breakpoint waits for an explicit signal from the target processor which indicates that the target processor has completed saving the state and to continue with the breakpoint handling.

13. Takase teaches a module that receives a signal that a breakpoint has occurred, (col. 8, line 48 to col. 9, line 19 describe the breakpoint and the signal that indicates a breakpoint has occurred). The state of the module is first saved (col. 9, lines 23-26), and then it outputs a stop notice signal, which indicates the module is stopped and the state has been kept, and then the debugger continues with the execution of the breakpoint. (Column 9, lines 20-42.) One of ordinary skill in the art would have recognized that using explicit signals, as taught in Takase, ensures the system is in a known, stable, and paused state before entering into the debugging, which, in turn,

would cause accurate debugging of the system and prevent altering of data prior to it being saved.

14. It would have been obvious to one of ordinary skill in the art to have a signal to indicate that the state has finished being saved. Fowler teaches a synchronized breakpoint system and debugging each of the plurality of processors during a breakpoint. One of ordinary skill in the art would have recognized that ensuring the system is in a known, stable, and paused state before entering into the debugging operations would cause accurate debugging of the system and prevent altering of data prior to it being saved. This can be ensured using an explicit signal which indicates the processors are in a stopped state and their state is saved, such as that taught by Takase.

15. As per claim 2, Fowler in view of Takase, teaches the method of claim 1 comprising resuming normal execution of the processor in response to a signal indicating that the saved state has been restored. (Step 10, which follows the restoring of the state of the peripheral in step 8, includes releasing a pause-out condition. Following the release of the pause-out condition, the utility processor returns to normal operation. Normally the utility processor has interrupts enabled, but they are disabled for the breakpoint handling, they are re-enabled in step 10, and hence, the utility processor resumes normal operation. Column 9, line 42-65)

16. As per claim 3, Fowler in view of Takase, teaches the method of claim 1 comprising resuming normal execution of the peripheral in response to a signal indicating that the processor has resumed normal execution: (Normally the utility

processor has interrupts enabled, but they are disabled for the breakpoint handling, they are re-enabled in step 10, and hence, the utility processor resumes normal operation. After the normal operation of the processor has resumed, indicated by the pause-in signal going inactive, the target subsystem resumes normal execution. Steps 10-15 in Column 9, line 42-65, also column 9, lines 23-31)

17. As per claim 4, Fowler in view of Takase, teaches the method of claim 1 comprising setting a register to control whether generation of the second signal is to be based on the state of the peripheral. (Fowler teaches in figures 6 and 7, column 5 line 63 to 68 and column 6, lines 25-32 a disable/enable switch, which has two positions that designate whether or not the target subsystem will be synchronized, i.e., paused for testing (breakpoint). Register is defined as "a device for storing small amounts of data; esp: one in which data can be both stored and operated on." (Merriam-Webster's Collegiate Dictionary, 10th Ed.) The switch is operated on and it also stores data (the current state, enabled or disabled). In the case of it being disabled, there would in turn be no second signal generation because the target subsystem would not be paused in the first place).

18. As per claim 5, Fowler in view of Takase, teaches the method of claim 1 comprising setting a register to control whether generation of a signal indicating that the saved state has been restored is to be based on the state of the peripheral. (Fowler teaches in figures 6 and 7, column 5 line 63 to 68 and column 6, lines 25-32 a disable/enable switch, which has two positions that designate whether or not the target subsystem will be synchronized, i.e., paused for testing (breakpoint) and then restored.

Register is defined as "a device for storing small amounts of data; esp: one in which data can be both stored and operated on." (Merriam-Webster's Collegiate Dictionary, 10th Ed.) The switch is operated on and it also stores data (the current state, enabled or disabled). In the case of it being disabled, there would in turn be no signal generation indicating the system had been restored because the target subsystem would not be paused in the first place. If the switch is storing an enable state, then Step 8, which follows the restoring the state of the peripheral step 7, includes releasing a pause-out condition, a signal to indicate the peripheral restored its state.)

19. As per claim 6, Fowler in view of Takase, teaches the method of claim 1 comprising triggering the breakpoint in response to a condition associated with occurrence of an instruction being executed by the processor. (Fowler teaches that the test utility system provides a breakpoint feature in order to halt/pause execution of a target subsystem and to proceed with testing and/or modifying the subsystem. Breakpoint is defined "to be set when both a point in the program and an event that will cause the suspension of execution at that point are defined" (The Authoritative Dictionary of IEEE Standards Terms, 7th Ed.) It is therefore inherent that when the test utility system implements the breakpoint feature in order to halt execution and test, it is when both a point in the program (occurrence of an instruction) and an event (condition) associated with the instruction has occurred.

20. As per claim 7, Fowler teaches a system comprising:

-A processor (Test utility system); a first computer-readable medium storing instructions that, when applied to the processor, cause the processor to: (The Test utility

system contains software to run a routine (Column 2, line 59 to column 3, line 17 and column 9, line 14 to column 10, line 22), it is inherent that the instructions in the program are stored on a computer-readable medium or else the computer would not be able to read them)

-Generate a first signal indicating execution of a breakpoint: (A breakpoint causes the target subsystem to halt execution, which is done via a pause/resume signal. (Steps 1-4, Column 9, lines 42-65, Column 1, lines 18-31 and column 9, lines 14-39))

-Continue execution of the breakpoint after the state of the peripheral has been saved: (Step 5 in Column 9, line 66 to Column 10, line 22 indicates the peripheral's state has been saved, and afterwards in step 6, the utility, in response to the saved state being completed, continues execution by carrying out the appropriate action, which as stated in column 2, lines 6-25 step 3 includes means to display and modify registers and column 3, lines 7-12 and column 1, lines 26-31. Since step 6 is the next sequential step in the synchronized process in Column 9, line 42-65 and includes modifying registers, it is inherent the saving of the state of the peripheral must be complete in step 5 before step 6 is started)

-And generate a third signal indicating that execution of the breakpoint has been completed: (The Test Utility Processor initiates a return to the application program in step 7 (shown in column 9, lines 42-65) via "an assert resume" portion of its software which also indicates the breakpoint has been completed. In the next step (step 8), after

the initiation in step 7, the state of the subsystem is restored. (Column 2, lines 66-68 and Column 9, lines 42-65).

-A peripheral coupled to the processor: (Target subsystem)

A second computer-readable medium storing instructions that, when applied to the peripheral, cause the peripheral to:

-Suspend execution and save a state of the peripheral, in response to receiving the first signal: (Step 5 in column 9, lines 42-65; Column 1, lines 26-31, Column 2, lines 59 to column 3, line 12)

-Restore the state of the peripheral, in response to receiving the third signal: (The Test Utility Processor initiates a return to the application program in step 7 (shown in column 9, lines 42-65) via "an assert resume" portion of its software which also indicates the breakpoint has been completed. In the next step (step 8), after the initiation in step 7, the state of the subsystem is restored. (Column 2, lines 66-68 and Column 9, lines 42-65).

Since the target subsystem does carry out the above steps, it is inherent that it is a result of instructions that are stored on a computer readable medium. A target subsystem inherently needs instructions to do any functions, and in order for the instructions to be read by the target subsystem, they are inherently stored on a computer readable medium.

21. While Fowler teaches that the context/state of the target processor is saved after a breakpoint and before each individual processor is debugged/monitored, Fowler does not teach that the processor which initiates the breakpoint waits for an explicit signal

from the target processor which indicates that the target processor has completed saving the state and to continue with the breakpoint handling.

22. Takase teaches a module that receives a signal that a breakpoint has occurred, (col. 8, line 48 to col. 9, line 19 describe the breakpoint and the signal that indicates a breakpoint has occurred). The state of the module is first saved (col. 9, lines 23-26), and then it outputs a stop notice signal, which indicates the module is stopped and the state has been kept, and then the debugger continues with the execution of the breakpoint. (Column 9, lines 20-42.)

23. It would have been obvious to one of ordinary skill in the art to have a signal to indicate that the state has finished being saved. Fowler teaches a synchronized breakpoint system and debugging each of the plurality of processors during a breakpoint. Ensuring the system is in a known, stable, and paused state before entering into the debugging operations would cause accurate debugging of the system and prevent altering of data prior to it being saved. This can be ensured using an explicit signal which indicates the processors are in a stopped state and their state is saved, such as that taught by Takase.

24. As per claim 8, Fowler, in view of Takase, teaches the system of claim 7 wherein the first computer-readable medium includes instructions that cause the processor to:

-Resume normal execution in response to receiving a fourth signal: (Step 9, which follows the restoring of the state of the peripheral (step 8), includes releasing a pause-out condition (fourth signal). Following the release of the pause-out condition, the utility processor returns to normal operation, step 10. Normally the utility processor

has interrupts enabled, but they are disabled for the breakpoint handling, they are re-enabled in step 10, and hence, the utility processor resumes normal operation. Column 9, line 42-65)

-And generate a fifth signal indicating that the processor has resumed normal execution: (Normally the utility processor has interrupts enabled, but they are disabled for the breakpoint handling, they are re-enabled in step 10, and hence, the utility processor resumes normal operation. After the processor (test utility) resumes normal operation, it indicates to the target subsystem (peripheral) to restart execution, which inherently involves a signal being generated and which is also an indication that the processor (test utility) has resumed normal operation. Steps 11-15 in Column 9, line 66 to column 10, line 22, also column 9, lines 23-31)

-And wherein the second computer-readable medium includes instructions that cause the peripheral to resume normal execution, in response to receiving the fifth signal: (After the processor (test utility) resumes normal operation, it indicates to the target subsystem (peripheral) to restart execution, which inherently involves a signal being generated and which is also an indication that the processor (test utility) has resumed normal operation. Steps 11-15 in Column 9, line 66 to column 10, line 22, also column 9, lines 23-31))

-And wherein the digital logic circuit is configured to generate the fourth signal indicating that the saved state of the peripheral has been restored: (Figure 8 depicts aspects of the digital logic circuit that generates the fourth signal, pause-out)

25. As per claim 9, Fowler, in view of Takase, teaches the system of claim 8, further comprising:

-A second processor (Another subsystem with both pause-in/resume-out signals and pause-out/resume-in signals in figure 1); a third computer-readable medium storing instructions that, when applied to the second processor, cause the second processor to:

-Suspend execution and save a state of the second processor, in response to receiving the first signal (pause-in): (Steps 1-4, column 9, line 66 to column 10, line 22, Column 1, lines 18-31 and column 9, lines 14-39))

-Restore the state of the second processor in response to receiving the third signal: (Column 9, lines 36-39, resume-in signal, steps 7)

-And resume normal execution, in response to receiving the fifth signal (pause-in signal): (Steps 11-14)

-And generate the fourth signal (pause-out signal) indicating that the saved state of the second processor has been restored: (Step 8, figure 8 depicts the interface circuitry generating the pause-out signal)

Since the second processor does carry out the above steps, it is inherent that it is a result of instructions that are stored on a computer readable medium. A processor inherently needs instructions to do any functions, and in order for the instructions to be read by the processor, they are inherently stored on a computer readable medium.

26. While Fowler teaches that the context/state of the target processor is saved after a breakpoint and before each individual processor is debugged/monitored, Fowler does not teach that the processor which initiates the breakpoint waits for an explicit signal

from the target processor which indicates that the target processor has completed saving the state and to continue with the breakpoint handling.

27. Takase teaches a module that receives a signal that a breakpoint has occurred, (col. 8, line 48 to col. 9, line 19 describe the breakpoint and the signal that indicates a breakpoint has occurred). The state of the module is first saved (col. 9, lines 23-26), and then it outputs a stop notice signal, which indicates the module is stopped and the state has been kept, and then the debugger continues with the execution of the breakpoint. (Column 9, lines 20-42.)

28. It would have been obvious to one of ordinary skill in the art to have a signal to indicate that the state has finished being saved. Fowler teaches a synchronized breakpoint system and debugging each of the plurality of processors during a breakpoint. Ensuring the system is in a known, stable, and paused state before entering into the debugging operations would cause accurate debugging of the system and prevent altering of data prior to it being saved. This can be ensured using an explicit signal which indicates the processors are in a stopped state and their state is saved, such as that taught by Takase.

29. As per claim 10, Fowler, in view of Takase, teaches the system of claim 7 including a system on a chip (SOC). (It is inherent that hardware of Fowler, in view of Takase, is constructed on a chip, and therefore, Fowler, in view of Takase, teaches a system (any of a collection of hardware taught in Fowler, in view of Takase) on a chip.)

30. As per claim 11, Fowler, in view of Takase teaches the system of claim 7 including a debugging tool (support processor 31) coupled to the system to debug the system. (Figures 1 and 3, column 4, lines 35-48)

31. As per claim 12, Fowler, in view of Takase, teaches the system of claim 7 wherein the digital logic circuit comprises a register to control whether generation of the second signal is to be based on the state of the peripheral: ((Fowler teaches in figures 6 and 7, column 5 line 63 to 68 and column 6, lines 25-32 a disable/enable switch, which has two positions that designate whether or not the target subsystem will be synchronized, i.e., paused for testing (breakpoint) and then restored. Register is defined as "a device for storing small amounts of data; esp: one in which data can be both stored and operated on." (Merriam-Webster's Collegiate Dictionary, 10th Ed.) The switch is operated on and it also stores data (the current state, enabled or disabled). In the case of it being disabled, there would in turn be no signal generation indicating the system had been restored because the target subsystem would not be paused in the first place. If the switch is storing an enable state, then Step 8, which follows the restoring the state of the peripheral step 7, includes releasing a pause-out condition, a signal to indicate the peripheral restored its state. Figures 6 and 7 are part of the interface circuit, as is figure 8, Column 6, lines 25-68)

32. As per claim 13, Fowler, in view of Takase, teaches the system of claim 7 wherein the state identifies a state of internal registers associated with the peripheral. (Column 1, lines 18-31, Column 2, lines 7-25, Column 9, lines 42-65)

33. As per claim 14, Fowler, in view of Takase, teaches the system of claim 7 wherein the processor operates at a clock rate different than the peripheral: (Column 1, lines 18-39)

34. As per claims 15, 17 and 23, given the similarities between claim 7 and claims 15, 17 and 23, the arguments as stated for the rejection of claim 7 also apply to claims 15, 17 and 23.

35. As per claims 16, 18 and 24, given the similarities between claim 8 and claims 16, 18 and 24, the arguments as stated for the rejection of claim 8 also apply to claims 16, 18 and 24.

36. As per claims 19, 21 and 25, given the similarities between claim 7 and claims 19, 21 and 25, the arguments as stated for the rejection of claim 7 also apply to claims 19, 21 and 25.

37. As per claims 20, 22 and 26, given the similarities between claim 8 and claims 20, 22 and 26, the arguments as stated for the rejection of claim 8 also apply to claims 20, 22 and 26.

38. As per claim 27, Fowler, in view of Takase, teaches the method of claim 1, wherein saving the state of the peripheral comprises saving the state of an input/output device. (Fowler teaches that the peripheral (target subsystem) is a processor. [Col. 2, lines 3-9] The processor has inputs and outputs [Figs. 1-4], and therefore is an input/output device, and its state is saved. [Column 1, lines 26-31, Column 2, lines 59 to column 3, line 12, Steps 1-4 in Column 9, line 42-65])

39. As per claim 29, Fowler, in view of Takase, fails to teach the method of claim 1, wherein suspending execution of the peripheral comprises suspending execution of a peripheral that is monolithically fabricated on a same chip as the processor.

40. It would have been obvious to one of ordinary skill in the art at the time the invention was made to make the multiple processors be integrated onto one monolithically fabricated chip since it has been held "that the use of a one piece construction instead of the structure disclosed in [the prior art] would be merely a matter of obvious engineering choice." [In re Larson, 340 F.2d 965, 968, 144 USPQ 347, 349 (CCPA 1965)]

41. Furthermore, Examiner takes Official Notice that putting hardware on a single, monolithically fabricated chip instead of separate chips for different pieces of hardware is well known and conventional in the art.

42. It would have been obvious to one of ordinary skill in the art at the time the invention was made to have the peripherals of Fowler, in view of Takase, on a single, monolithically fabricated chip, since Examiner takes Official Notice this is a well known and conventional technique in the art.

43. As per claims 30, 33, 36 and 40, given the similarities between claim 27 and claims 30, 33, 36 and 40, the arguments as stated for the rejection of claim 27 also apply to claims 30, 33, 36 and 40.

44. As per claims 32, 35, 38, 39 and 42, given the similarities between claim 29 and claims 32, 35, 38, 39 and 42, the arguments as stated for the rejection of claim 29 also apply to claims 32, 35, 38, 39 and 42.

45. Claims 28, 31, 34, 37 and 41 are rejected under 35 U.S.C. 103(a) as being unpatentable over Fowler, U.S. Patent 4,502,116 in view of Takase, U.S. Patent 6,289,473 and further in view of Hicok 5,561,761.

46. As per claim 28, Fowler, in view of Takase, teaches the method of claim 27, however fails to further teach wherein saving the state of the input/output device comprises saving the state of a universal asynchronous receiver/transmitter (UART).

47. Hicok teaches that when debugging a processor using a breakpoint in a multiprocessor environment, it is advantageous to interrogate the elements of the individual processor, including an input/output device (Data Buffer 18 and Data Buffer 20, fig. 1). It would have been obvious to one of ordinary skill in the art to include an input/output device on a processor because, as one of ordinary skill in the art would have recognized, it allows the processor to communicate with external devices and users. The system taught by Hicok allows a non-destructive access, i.e., the state is saved when the CPU is stopped. [Col. 1, lines 55-63 and col. 2, lines 39-59] It would have been obvious to include the ability to debug an input/output device within a processor in a multiprocessor environment to the system of Fowler and Takase because it allows non-destructive interrogation. Therefore, it would have been obvious to include the input/output device and the method of interrogating it, as taught by Hicok, into the system of Fowler and Takase.

48. However, Fowler, in view of Takase and Hicok, does not specify on the hardware to implement the input and output device.

49. Examiner takes Official Notice that it is well known in the art to implement an Input/Output device on a processor as a UART in order to advantageously provide an asynchronous means to communicate with external devices.

50. It would have been obvious to one of ordinary skill in the art at the time the invention was made to specify the Input/Output devices of Hicok as UARTs since Examiner takes Official Notice asynchronous communication via UARTs is a well known an advantageous method to communicate with external devices.

51. As per claims 31, 34, 37 and 41, given the similarities between claim 28 and claims 31, 34, 37 and 41, the arguments as stated for the rejection of claim 28 also apply to claims 31, 34, 37 and 41.

Response to Arguments

52. Applicant's arguments, see Pre-Appeal Brief, filed 1/9/06, with respect to the rejection(s) of claim(s) 1-42 under 35 U.S.C. 103 have been fully considered and are persuasive. Therefore, the rejections have been withdrawn. However, upon further consideration, a new ground(s) of rejection is made above. Furthermore, Applicant's arguments filed on 1/9/2006 regarding Fowler (U.S. Patent 4,502,116), and shown below, have been fully considered but they are not found persuasive.

53. Applicant argues the novelty/rejection of claims 1, 7, 15, 17 and 23.

> "Fowler specifies that his interface circuit handles four different types of signals, i.e., pause-in signals, pause-out signals, resume-in signals, and resume-out signals. None of these signals indicates anything about the save state of another processor. Fowler thus teaches that any instruction execution is independent of the save state of other processors in his multiprocessor system. This stands in direct contrast to claims 1, 7, 15, 17 and 23, in which execution of a breakpoint is continued in response to receipt of a signal indicating that the state of a peripheral has been saved."

54. These arguments are not found persuasive for the following reasons:

a. The fact that Fowler fails to teach a signal that indicates a save state of the target processor does not inherently mean that the continuing execution of the breakpoint is independent of said save state. There is no explicit teaching by Fowler that the continued execution of the breakpoint is independent or dependent of the save state of the target processor, Fowler is silent on this feature. Therefore Examiner disagrees with the inference that "Fowler thus teaches that any instruction execution is independent of the save state of other processors in his multiprocessor system."

Conclusion

The following is text cited from 37 CFR 1.111(c): In amending in reply to a rejection of claims in an application or patent under reexamination, the applicant or patent owner must clearly point out the patentable novelty which he or she thinks the claims present in view of the state of the art disclosed by the references cited or the objections made. The applicant or patent owner must also show how the amendments avoid such references or objections.

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the

shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Kevin P Rizzuto whose telephone number is (571) 272-4174. The examiner can normally be reached on M-F, 8-4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

KPR

Eddie Chan

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100